Computers and Mathematics with Applications 59 (2010) 779–786

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Detection of over-represented motifs corresponding to known TFBSs via motif clustering and matching

Liu Li-fang [a,b,*], Jiao Li-cheng [b]

[a] School of Computer Science and Technology, Xidian University, Xi'an 710071, China
[b] Institute of Intelligent Information Processing, Xidian University, Xi'an 710071, China

ARTICLE INFO

Article history:
Received 24 July 2008
Received in revised form 7 October 2009
Accepted 7 October 2009

Keywords:
Motif discovery
Binding sites
Transcription factors
p-value
Statistical significance

ABSTRACT

Detection of over-represented motifs corresponding to known TFBSs (Transcription Factor Binding Sites) is an important problem in biological sequences analysis. In this paper, a novel motif discovery method based on motif clustering and matching is proposed. Against a precompiled library of motifs described as position weight matrices (PWMs), each *L*-mer in the data set is matched to a motif base on the match score's *p*-value, and then the PWMs are updated and clustered according to their similarity. Motif features are ranked in terms of statistical significance (*p*-value). We present an implementation of this approach, named MotifCM, which is capable of discovering multiple distinct motifs present in a single data set. We apply our method to the benchmark which has 56 data sets, and demonstrate that the performance of MotifCM on this data set compares well to, and in many cases exceeds, the performance of existing tools.

1. Introduction

Transcription factors are biomolecules that bind to short segments of DNA. These short segments, which are sometimes referred to as motifs, have a conserved appearance. Finding shared motifs in DNA sequence remains a fundamental problem in computational biology. Over the past few years, numerous tools have become available for this task of motif discovery (for reviews see [1–3]). Nearly all motif discovery algorithms fall into three general classes: pattern-based, profile-based and combinatorial. In profile-based algorithms, a motif is usually modeled by a $4 \times L$ position weight matrix (PWM), where L is the motif's length in base pairs, so that each column of the PWM represents the distribution of the 4 nucleotide types at the corresponding position in the motif. One way of estimating the PWM can be through standard statistical learning theory methods, such as maximum-likelihood estimation (e.g. MEME [4], The Improbizer [5]), Markov chain Monte Carlo algorithms (e.g. AlignACE [6], BioProspector [7], MotifSampler [8], GLAM [9], DSMC [10]), and greedy search (e.g. Consensus [11]). Benchmark experiments [1,2] of some of these motif discovery tools reveal that the nucleotide and the binding site level prediction accuracy are very low in DNA sequences, so existing methods are still in need of much improvement. Another way of finding shared motifs is to compile a library of motifs which were previously characterized, such as JASPAR [12] and TRANSFAC [13], and assess whether any of these motifs are statistically over-represented in the sequences (e.g. Clover [14]). Library-based methods have shown improved performance. There are also many alternative motif identification methods (e.g. SOMBRERO [15], PHYLOCLUS [16], Lones et al. [17]). These methods use clustering to group similar patterns. SOMBRERO uses the clustering properties of the self-organizing map to exhaustively characterize all motif features present within a

* Corresponding author at: School of Computer Science and Technology, Xidian University, Xi'an 710071, China.
E-mail address: xdliulifang@gmail.com (L.-f. Liu).

0898-1221/$ – see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2009.10.016

L-mer	CCACGG					
Motif	1	2	3	4	5	6
A	1.5	0.7	**1.8**	1.6	1.6	0.8
C	**0.2**	**1.4**	-3.6	**-3.8**	-3.9	-3.6
G	-4.0	-4.0	-4.2	-1.6	**-1.4**	**-0.8**
T	-4.1	-2.3	-3.9	-2.0	-1.4	0.2
Match Score	S = 0.2+1.4+1.8-3.8-1.4-0.8 = -2.6					

Fig. 1. Computing motif match score.

set of input sequence. PHYLOCLUS first uses a *de novo* motif discovering method to find motifs and then uses a Bayesian hierarchical clustering method to cluster those motifs. The method described by Lones et al. uses an evolutionary algorithm and the algorithm uses data clustering to logically distribute the evolving population across the search space.

In this work, we present a novel approach to the *de novo* identification of conserved motifs in biological sequences based on motif clustering and matching. Based on a precompiled library of motifs that represented by PWMs which are either random or prior initialized, each L-mer in the search space is matched to a motif according to the match score's p-value, and the L-mer becomes a motif instance. Once all L-mers have been matched, the PWMs in the library should be updated based on the matched motif instances. This, however, might result in a redundant set of PWMs, so similar motifs should be removed or merged into a new PWM. Thus motif clustering is applied. In order to identify motifs that are over-represented as having a functional regulatory role, the PWMs are ranked in terms of statistical significance (p-value) relative to a background model.

The advantage of this approach is that it can be used to simultaneously characterize every feature present in the data set thus lessening the chance that weaker signals will be missed. Our approach is demonstrated here using an implementation named MotifCM (motif discovery via motif clustering and matching). MotifCM's performance is evaluated here in comparison with several other popular motif discovery tools described by Tompa et al. [1].

2. Materials and methods

2.1. Method for motif matching

According to the match score's p-value, each L-mer in the data set is matched to a motif within the motif library. The match score of a motif is defined as the sum of the scores for the letters in the L-mer matched with columns 1 to L of the motif, respectively. This is illustrated in Fig. 1. The p-value of the match between a motif and an L-mer is defined as the probability of observing a match score at least as good when the motif is matched to a random L-mer.

Following Bailey, and Gribskov [18], we obtain the match score's p-value by calculating the cumulative density function. Define $m_{j,k}$, $1 \le j \le 4$, $1 \le k \le L$ as the PWM's entries. The null hypothesis assumes that each position in a sequence is i.i.d. with the average letter distribution observed in naturally occurring sequences, q_i, $1 \le i \le 4$. Let $M^{(k)}(x)$ be the match score probability density function for the motif matrix if it consisted of only its first k columns, then the density for the matrix consisting of the first $k+1$ columns is

$$M^{(k+1)}(x) = \sum_{j=1}^{4} M^{(k)}(x - m_{j,k+1}) q_j. \tag{1}$$

After L iterations, $M^{(L)}(x)$ contains the probability density for matching the motif with a random L-mer, from which the match score's p-value is

$$P(s) = \sum_{x \ge s} M^{(L)}(x). \tag{2}$$

An L-mer is matched to a motif which has the smallest match score's p-value.

2.2. Method for motif clustering

Methods for comparing motifs have been described by Pietrokovski [19], Schones et al. [20], and Pape et al. [21]. Here we chose a scoring method based on the Pearson correlation coefficient between the nucleotide base frequencies of two motif alignments. Similarity measure between two columns X and Y can be written as in Eq. (3).

$$PCC(X, Y) = \frac{\sum_{b=A}^{T} (X_b - \bar{X})(Y_b - \bar{Y})}{\sqrt{\sum_{b=A}^{T} (X_b - \bar{X})^2 \sum_{b=A}^{T} (Y_b - \bar{Y})^2}} \tag{3}$$

where X_b and Y_b are the probability values of base b in columns X and Y, respectively, and $\bar{X}$ and $\bar{Y}$ are the means of the values in columns X and Y, respectively. This score varies between -1 and 1, and approaches 1 for a perfect match between the two columns. To compare matrices consisting of multiple columns, the scores of the individual column comparisons are averaged.

Here, agglomerative hierarchical clustering is used for motif clustering. The clustering procedure consists of five main steps. The clustering algorithm is described as follows:

Algorithm: MotifCM clustering algorithm
Initialization: Compute an all-against-all similarity matrix based on PWMs in library.
Repeat
1. Select the two most similar elements with maximum score.
2. Merge the two selected elements and update the similarity matrix scores by remaining the smallest similarity between the motifs in the newly created cluster and all other motifs.

Until An appropriate condition is met (e.g. there is no members sharing sufficient similarity (average $PCC < 0.7$)).
Output: Create new PWMs from the alignments of the clustered motifs' instances.

2.3. Motifs ranking and identifying

The motif library will contain different motifs present in the input sequences. The various motifs found by MotifCM must be ranked in terms of over-representation against a stochastic model of occurrence. The p-value of a motif score is used here for the purpose of ranking the motifs. The evaluation of un-gapped local alignment is usually made using its information content or relative entropy [22]:

$$I = \sum_{i=1}^{L} \sum_{j=1}^{|\Omega|} n_{i,j} \log \frac{n_{i,j}/N}{q_j} \tag{4}$$

where $\Omega = [A, C, G, T]$, $n_{i,j}$ count of the jth letter in the ith column of alignment, N is the number of subsequences in the alignment and q_j the background frequency of the jth letters. Using this scoring function (4) and a null model, which assumes that each of the L columns has N letters independently sampled according to the background distribution we can estimate a p-value. The p-value for a given scoring value s represents the probability of an entropy score of s or better under the null model. We use the method described in MEME to calculate the motif score's p-value.

In order to filter certain repetitive "simple" genomic features (such as poly (A) sequences or repeats), a complexity filter is employed. The complexity score [15,23] is given by

$$C(\text{alignment}) = \left(\frac{1}{4}\right)^{L} \prod_{b=A}^{T} \left(\frac{L}{\sum_{i=1}^{L} p_{i,b}} \right)^{\sum_{i=1}^{L} p_{i,b}} \tag{5}$$

where $p_{i,b}$ is the frequency of the base b in the ith column of the alignment. Any alignment which receives less than a reasonably low complexity score (0.15 is used in this study) is discounted from being treated as a possible functional motif.

2.4. Description of MotifCM

In the current application of MotifCM, two initialization strategies are used for building the precompiled motif library. Randomly select a sequence from the data set, if the length of the selected sequence is S, the motif library's length is $S-L+1$. The first method for initializing the PWMs in the library is the standard random initialization. The second method is 'prior initialization', a strategy to built models from L-mers of the randomly selected sequences for building the library. Each L-mer of the selected sequence is associated with a position frequency matrix θ that is constructed as in Eq. (6).

$$\theta = \left[p_{i,j}\right], \quad \text{where } p_{i,j} = \begin{cases} \lambda & \text{if } a_{i,j} = \alpha_j \\ \dfrac{(1-\lambda)}{(|\Omega|-1)} & \text{otherwise} \end{cases} \tag{6}$$

where $\alpha_j \in \Omega$, $\lambda > 1/|\Omega|$, then a PWM $m_{i,j} = \log_2(p_{i,j}/q_j)$ is also defined. The frequency matrices are updated using Eq. (7).

$$p_{i,j} = \frac{n_{i,j} + \beta q_j}{N + \beta}. \tag{7}$$

Here, βq_j is an unbiased pseudocount, in proportion to the background frequency q_j, and scaled down by the factor $\beta = 0.1$. During the iteration procedure, $p_{i,j}$ is also exponentiated by a Boltzmann temperature factor kT, $\hat{p}_{i,j} = p_{i,j}^{1/kT}$, kT is started at 5 and reduced by 5% at each iteration, if kT cooled below 0.1, updating is reverted to Eq. (7). In conclusion, the MotifCM is

described as follows:

Algorithm: MotifCM algorithm
Initialization: Building precompiled motif library using one of the two initialization strategies. Let $kT = 5$.
Repeat
1. Matching each L-mer in the data set to a motif according to the match score's p-value, as described in Section 2.1.
2. Updating the PWMs in motif library.
if ($kT < 0.1$)
update parameters in PWMs using Eq. (7).
else
update parameters in PWMs using an annealing schedule $\hat{p}_{i,j} = p_{i,j}^{1/kT}$, as described above.
$kT = kT * 95.0/100$.
3. Clustering the PWMs, as described in Section 2.2.
Until an appropriate condition is met (e.g. a specified number of training iterations are completed, default 500, or clustering operation has not executed for some continuous number of iterations, default 10).
Output: Ranking the motifs according to the motif's p-value, as described in Section 2.3. The top 10 predictions are printed.

3. Results

3.1. Performance assessment

We use the nucleotide and site level accuracy [1,2] to evaluate the performance of our algorithm. The following values for calculating accuracy metrics are defined: *nTP* (true positive), the number of nucleotide positions in both known sites and predicted sites; *nFN* (false negative), the number of nucleotide positions in known sites but not in predicted sites; *nFP* (false positive), the number of nucleotide positions not in known sites but in predicted sites; *nTN* (true negative), the number of nucleotide positions in neither known sites nor predicted sites; *sTP*, the number of known sites overlapped by predicted sites; *sFN*, the number of known sites not overlapped by predicted sites; *sFP*, the number of predicted sites not overlapped by known sites. Then, at either the nucleotide ($x = n$) or site ($x = s$) level, one can define the sensitivity xS_n:

$$xS_n = xTP/(xTP + xFN), \tag{8}$$

and the positive predictive value *xPPV*:

$$xPPV = xTP/(xTP + xFP). \tag{9}$$

In order to capture both specificity and sensitivity in a single accuracy measurement, the nucleotide level performance coefficient is define as:

$$nPC = nTP/(nTP + nFN + nFP), \tag{10}$$

the nucleotide level correlation coefficient is define as:

$$nCC = \frac{(nTP \cdot nTN - nFN \cdot nFP)}{\sqrt{(nTP + nFN)(nTN + nFP)(nTP + nFP)(nTN + nFN)}} \tag{11}$$

and the site level average site performance as:

$$sASP = (sS_n + sPPV)/2. \tag{12}$$

3.2. Case study I: The binding sites stored in TRANSFAC database

The data sets are available as a benchmark at the assessment web site http://bio.cs.washington.edu/assessment/. It contains 56 data sets with known binding sites from fly, human, mouse and yeast. The binding sites of each transcription factor were presented in three different background models, 'real' (the real promoter sequences), 'generic'(randomly chosen promoter sequences from the same genome), and 'Markov' (sequence generated by a Markov chain of order 3). Our results are compared to the 13 tools evaluated in Tompa et al. The comparative results of this evaluation are described below. For MotifCM, the motif width ranges from 6 to 18, the match score's p-value cutoff is 0.0001. The initialization strategy is 'prior initialization'. For the top 10 discovered motifs, only select the motif which overlaps the most positions with the real binding sites, and report the positions and sequences of that motif's occurrences. Fig. 2a shows the results of all the 56 data sets (regardless of species, data set type). Fig. 2b breaks down the data sets according to species (regardless of data set type) using *nCC* as a proxy for correctness. Fig. 2c breaks down the data sets according to type real, generic or Markov (regardless of species). Fig. 2d shows the results of *nTP* and *nFP*.

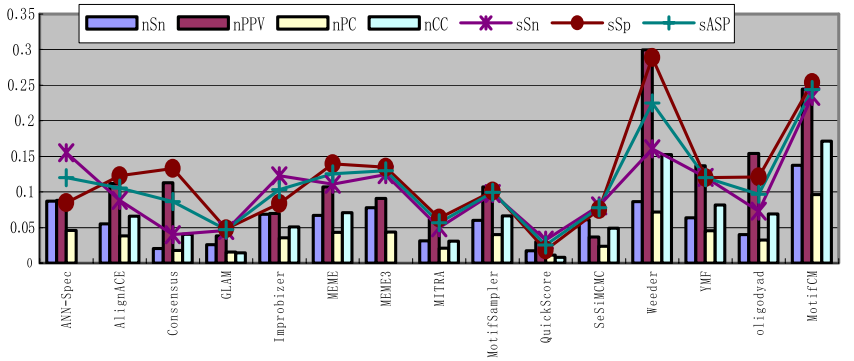

Fig. 2a. Performances over all 56 data sets.

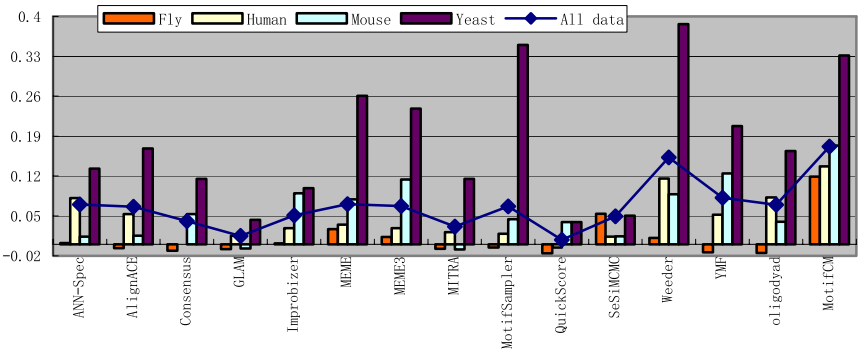

Fig. 2b. *nCC* by species.

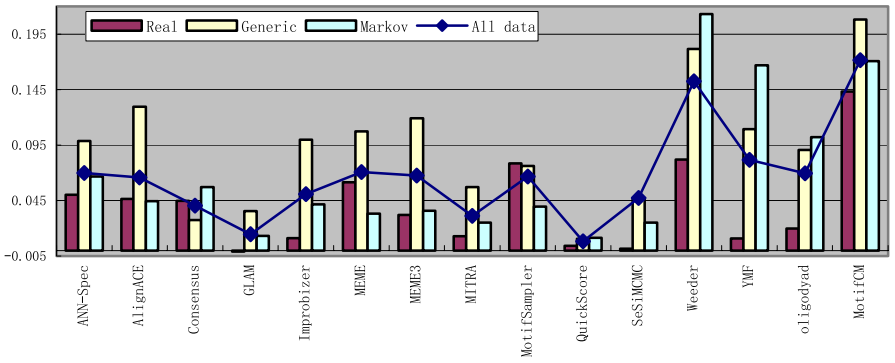

Fig. 2c. *nCC* by data set type.

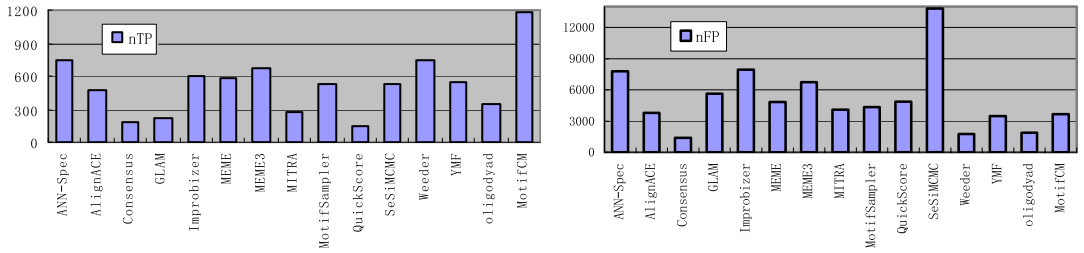

Fig. 2d. *nTP* and *nFP* over all 56 data sets.

From the results, we found that MotifCM achieves improved performance over the other popular motif discovery tools except for *nPPv* and *sSp* in which it falls behind Weeder. However, as explained in [1], the *nPPv* and *sSp* values tend to be

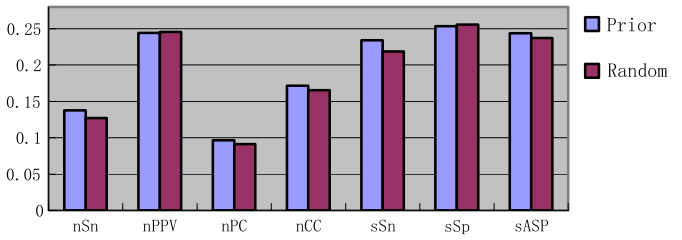

Fig. 3. MotifCM's performance using various initialization strategies.

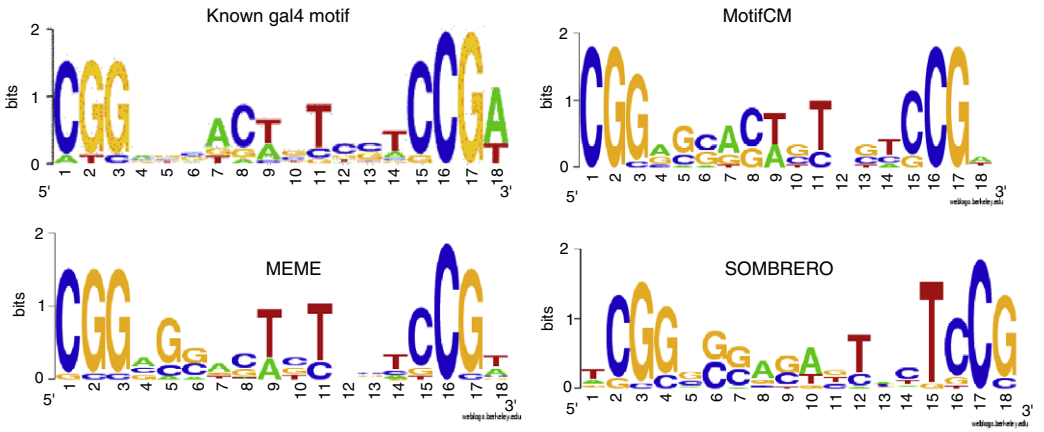

Fig. 4. Comparison of known logos to predicted logos.

exaggerated for those programs that make no predictions on data sets. MotifCM had a motif prediction on every data set. It attempted a prediction while Weeder on the other hand had 17 data sets which it predicted no motif. But we also found that at the nucleotide and site level, the prediction accuracy of all algorithms is relatively low. The accuracy levels are lower than the performance scores reported on ECRDB62A set [2]. This is due to their longer sequences ranging from 500 to 3000 nt, while the sequence lengths in ECRDB62A vary from 86 to 676 nt. This is also the reason why motifCM works better for yeast and less for fly, mouse and human. MotifCM performs better for 'generic' type sequence than for 'real' and 'Markov' type. That may be due to the reason that the background probability of the four nucleotides (A, C, G, T) used by MotifCM is generated from the whole genome. When p-value is calculated, the probability is more meaningful for 'generic' type sequences.

Here, the performances of two MotifCM initialization strategies are also compared when applied to the benchmark. As can be observed from Fig. 3, there is little variation between the average performance of the random initialization and the 'prior initialization'. When using random initialization, there is no substantial decrease in accuracy.

3.3. Case study II: Yeast genomic sequences taken from S. cerevisiae

The data set used in this example is a set of 6 promoter sequences taken from *S. cerevisiae*. The 6 sequences are known to harbour 14 binding sites for the gal4 transcription factor. There are 3100 bp in total in the data set. The data set are available at the web site http://bioinf.nuigalway.ie/sombrero. We compare the performance of MotifCM with two motif identification programs, MEME and SOMBRERO. MEME is run online (http://meme.sdsc.edu/) with arguments -mod tcm, -nmotifs 3, -minw 18, and -maxw 18. SOMBRERO result is taken from its web site. The motifs discovered by the three programs are shown as sequence logos in Fig. 4 below. The experimentally verified gal4 binding motif is also shown in Fig. 4. As you can see, MotifCM finds a close match to the gal4 binding motif.

In terms of how successful the three tools are at finding the binding site locations, we can compare the list of occurrences of the above motifs to the known gal4 binding sites, which shows that MotifCM correctly finds 12 of the 14 known gal4 binding sites, and no negative sites. MEME finds 13 of the known gal4 sites, and 3 other sites that are not known to be gal4 binding sites. SOMBRERO finds 13 of the known gal4 sites, and 4 other sites. The comparison of the nucleotide and site level accuracy of the three tools is listed in Fig. 5. The difference of the 7 values between MotifCM and MEME (or SOMBRERO) is not statistically significant, for the p-value of the Wilcoxon Matched-Pairs Signed-Ranks Test is 0.2188 (or 0.375). When we do motif matching, an L-mer is matched to a motif based on their match score's p-value. We considered an L-mer with a match score's p-value cutoff 0.0001 or less for the above evaluation. The value 0.0001 was selected because it yielded comparatively similar amounts of predicted motifs as the other tools analyzed. Reducing this value increases the quality of the predicted motifs, however, it reduces the number of motifs predicted. Increasing this value on the other hand results in more data sets with predicted motifs but with a lower average quality. When setting p-value cutoff 0.001, MotifCM correctly

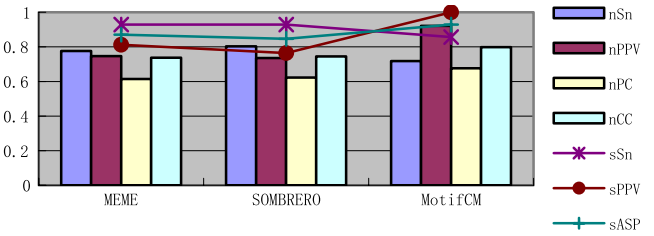

Fig. 5. Comparison of the nucleotide and site level accuracy.

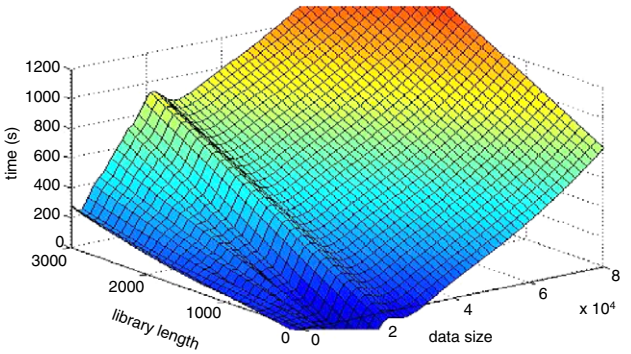

Fig. 6. Relation among execution time, data size and motif library length.

finds 13 of the 14 known gal4 binding sites, it also finds 5 other matches to the above motif that are not known to be gal4 binding sites. Similarly, when setting p-value cutoff 0.01, MotifCM correctly finds 14 of the 14 known gal4 binding sites, but the number of negative sites increased to 7. It is necessary to select an appropriate threshold value when the more accurate calculation method for matching score's p-value is needed if an improved performance for MotifCM can be achieved.

3.4. Speed of MotifCM

The execution time required for MotifCM is related to the data size and the motif library length. We noted the execution time of the 56 data sets from Tompa et al. on a HP Compaq dc7700 convertible Minitower (Intel® Core(TM)2 CPU, 1.86 GHz, 1.97 GB memory). Fig. 6 shows the relation among the execution time, the data size and the motif library length. From Fig. 6 we see that along with the increase of the data size and the motif library length, the execution time of MotifCM is also increased.

4. Discussion

We have developed a new motif discovery algorithm based on motif clustering and matching. It should be noted that our approach is quite different to probabilistic motif discovery. Like SOMBRERO, our method allows all motifs in a data set to be simultaneously characterized. We compare the performance of our method to those of MEME, AlignACE, MotifSampler, Consensus, GLAM, The Improbizer, and SOMBRERO et al., and the benchmark results in case study I show that MotifCM performs equal or better than the other tools except for Weeder. But in case study II, MotifCM does not perform signification better than the others. We note that MotifCM can achieve a comparable performance.

Our experiments indicate that the highest significance scores (we use the p-value) are not necessarily the best prediction of the target motifs, and this can lead to lower prediction accuracy. The lack of correlation between the significance scores and the accuracy scores shows that high significance score does not necessarily indicate high prediction accuracy. How to tackle the consistency between the significance scores and the accuracy scores is our next work.

MotifCM is based on motif clustering and matching, the key issue in this strategy is the calculation of the match score's p-value and the similarity measures between motifs. The efficient and accurate computation of p-value and similarity is the key to improving the performance of MotifCM. Future improvements to the clustering and matching based (library-based) motif finder could incorporate more accurate methods for estimating the p-value and similarity. Improvements to this method could also be obtained by incorporating similar probabilistic models in the algorithm as SOMBRERO and Clove do, thus allowing the application of the method to larger promoter sequences, especially those yielded by eukaryotic gene expression experiments.

Acknowledgement

This work was supported by the NNSF of China under Grant No. 60705004.

References

[1] M. Tompa, N. Li, T.L. Bailey, G.M. Chruch, B. De Moor, E. Eskin, et al., Assessing computational tools for the discovery of transcription factor binding sites, Nature Biotechnology 23 (2005) 137–144.
[2] Jianjun Hu, Bin Li, Daisuke Kihara, Limitations and potentials of current motif discovery algorithms, Nucleic Acids Research 33 (2005) 4899–4913.
[3] Geir Kjetil Sandve, Finn Drabløs, A survey of motif discovery methods in an integrated framework, Biology Direct 1 (11) (2006) http://www.biology-direct.com/content/1/1/11.
[4] T.L. Bailey, C. Elkan, Unsupervised learning of multiple motifs in biopolymers using expectation maximization, Machine Learning 21 (1995) 51–80.
[5] W. Ao, J. Gaudet, W.J. Kent, S. Muttumu, S.E. Mango, Environmentally induced foregut remodeling by PHA-4/FoxA and DAF-12/NHR, Science 305 (2004) 1743–1746.
[6] J.D. Hughes, P.W. Estep, S. Tavazoie, G.M. Church, Computational identification of *cis*-regulatory elements associated with functionally coherent groups of genes in Saccharomyeds cerevisiae, Journal of Molecular Biology 296 (2000) 1205–1214.
[7] X. Liu, D.L. Brutlag, J.S. Liu, BioProspector: Discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes, Pacific Symposium on Biocomputing 6 (2001) 127–138.
[8] G. Thijs, K. Marchal, M. Lescot, S. Rombauts, B. De Moor, P. Rouze, Y. Moreau, A Gibbs sampling method to detect overrepresented motifs in the upstream regions of co-expressed genes, Journal of Computational Biology 9 (2002) 447–464.
[9] M.C. Frith, U. Hansen, J.L. Spouge, Z. Weng, Finding functional sequence elements by multiple local alignment, Nucleic Acids Research 32 (2004) 189–200.
[10] K.C. Liang, X.D. Wang, D. Anastassiou, A profile-based deterministic sequential Monte Carlo algorithm for motif discovery, Bioinformatics 24 (2008) 46–55.
[11] G. Hertz, G. Stormo, Identifying DNA and protein patterns with statistically significant alignments of multiple sequences, Bioinformatics 15 (1999) 563–577.
[12] A. Sandelin, W. Alkema, P. Engstom, W. Wasserman, B. Lenhard, JASPAR: An open-access database for eukaryotic transcription factor binding profiles, Nucleic Acids Research 32 (2004) D91–D94.
[13] E. Wingender, X. Chen, R. Hehl, H. Karas, J. Liebich, V. Matys, T. Meinhardt, M. Pruss, I. Reuter, F. Schacherer, TRANSFAC: An integrated system for gene expression regulation, Nucleic Acids Research 28 (2000) 316–319.
[14] Martin C. Frith, Yutao Fu, Liqun Yu, et al., Detection of functional DNA motifs via statistical over-representation, Nucleic Acids Research 32 (2004) 1372–1381.
[15] S. Mahony, D. Hendrix, A. Golden, T.J. Smith, D.S. Rokhsar, Transcription factor binding site identification using the self-organizing map, Bioinformatics 21 (2005) 1807–1814.
[16] Shane T. Jensen, Lei Shen, Jun S. Liu, Combining phylogenetic motif discovery and motif clustering to predict co-regulated genes, Bioinformatics 21 (2005) 3832–3839.
[17] Michael A. Lones, Andy M. Tyrrell, Regulatory motif discovery using a population clustering evolutionary algorithm, IEEE/ACM Transactions on Computational Biology and Bioinformatics 41 (3) (2007) 403–414.
[18] Timothy L. Bailey, Michael Gribskov, Combining evidence using *p*-values: Application to sequence homology searches, Bioinformatics 14 (1998) 48–54.
[19] S. Pietrokovski, Searching databases of conserved sequence regions by aligning protein multiple-alignments, Nucleic Acids Research 24 (1996) 3836–3845.
[20] D.E. Schones, P. Sumazin, M.Q. Zhang, Similarity of position frequency matrices for transcription factor binding sites, Bioinformatics 21 (2005) 307–313.
[21] U.J. Pape, S. Rahmann, M. Vingron, Natural similarity measures between position frequency matrices with an application to clustering, Bioinformatics 24 (2008) 350–357.
[22] G.D. Stormo, DNA binding sites: Representation and discovery, Bioinformatics 16 (2000) 16–23.
[23] H. Wan, L. Li, J.C. Wootton, Discovering simple regions in biological sequences associated with scoring schemes, Journal of Computational Biology 10 (2003) 171–185.